

A swarm solution for a cooperative and self-organized team of UAVs to search targets

Patrícia de Sousa
Paula¹

Miguel Franklin de
Castro²

Gabriel A. Louis
Paillard³

Wellington W. F.
Sarmiento⁴

Universidade Federal do Ceará, UFC
Fortaleza, CE, Brasil

¹patricia@virtual.ufc.br, ²miguel@ufc.br, ³gabriel@virtual.ufc.br, ⁴wwagner@virtual.ufc.br

ABSTRACT

The context of this research is the use of Unmanned Aerial Vehicles (UAV) to search and localize a target. The problem to be solved is the cooperative behavior of the UAV group to optimize the search. The focus of this research is the use of bioinspired algorithms to spread the aerial vehicles and the optimization of the time of search, comparing the application of Particle Swarm Optimization with others found in literature.

Keywords

Unmanned Aerial Vehicle; UAV; swarm intelligence; UAV swarm; bio-inspired.

1. INTRODUÇÃO

O avanço do uso de veículos autônomos para aplicações fora de laboratórios de pesquisa vem aumentando consideravelmente nos últimos anos. Assim, os veículos autônomos estão se inserindo aos poucos em nosso dia a dia, sendo chamados UGV (Unmanned Ground Vehicle), UAV (Unmanned Aerial Vehicle) e MAV (Micro Aerial Vehicle), dependendo da movimentação e tamanho. Os UGVs, UAVs e MAV tem relação com a computação autônoma, podendo ser parcial ou completamente autônomos, de acordo com as características levantadas no referencial teórico a ser visto na seção 2.

Uma questão fundamental no uso de UAVs com WSN (*Wireless Sensor Networks*) é o comportamento e distribuição dos UAVs (ver Figura 1) para varredura de uma área. Tal comportamento pode ser modelado por meios mais tradicionais, como Máquinas de Estado Finito (no inglês Finite State Machine - FSM) ou por abordagens inspiradas em comportamentos de seres vivos, o que é chamado de Computação Bioinspirada [1][2].

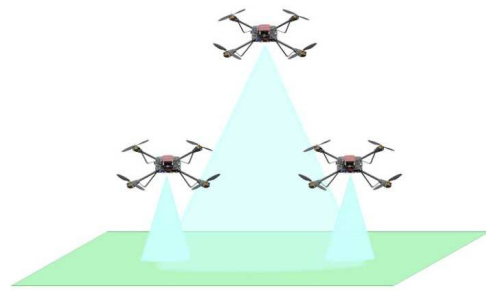


Figura 1 Ilustração de UAVs atuando em conjunto
[Fonte:<http://cabuavphd.blogspot.com.br/2012/11/project-progress-search-patterns-and.html>]

É neste contexto que se encontra o presente trabalho, o qual propõe uma solução bioinspirada para a distribuição e comportamento de UAVs para a identificação de pessoas ou veículos em ambientes abertos e/ou de difícil acesso (e.g. florestas, rede de cavernas e montanhas).

2. FUNDAMENTAÇÃO TEÓRICA

Algumas definições e tecnologias que contextualizam a presente pesquisa são colocadas a seguir.

2.1. Veículos Aéreos Não Tripulados

Os UAVs (Unmanned Aerial Vehicles) – ou, no português, Veículos Aéreos Não Tripulados (VANTs) –, são aeronaves sem a presença de pilotos que podem ser controladas remotamente ou voar de forma autônoma de acordo com planos de voo pré-programados [3][4]. Suas principais formas de controle são via rádio remoto ou pela pré-programação do voo.

Sua versão em tamanho reduzido é chamada MAV, Micro Aerial Vehicle. A versão terrestre é chamada Unmanned Ground Vehicle, UGV, que normalmente assume a forma de carros ou robôs com deslocamento terrestre. Os UAVs levam vantagem sobre os UGVs no que diz respeito ao terreno, que podem ser íngremes e/ou com muitos obstáculos, e no maior alcance de área, lembrando que missões de maior duração vão requerer que os UAVs retornem periodicamente para reabastecimento [5].

Os UAVs tem capacidade de sensoriamento limitada e a cooperação entre um grupo de UAVs depende fortemente da comunicação entre UAVs próximos. Como exemplo, pode-se ver na Figura 2 que os nós A e F não possuem um canal direto de comunicação. Para que a

informação seja enviada, deve-se seguir um caminho intermediário de comunicação, por exemplo, $A > B > C > F$. No caso dos nós UAVs estarem acoplados a nós de uma rede WSN, essa comunicação pode ser feita pela rede WSN, utilizando protocolo de comunicação próprio desse tipo de rede.

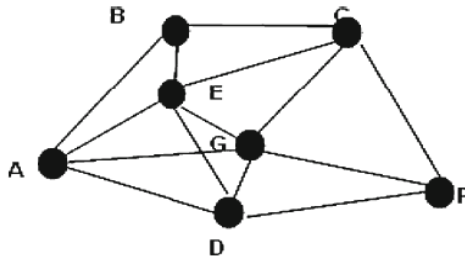


Figura 2 Exemplo de nós em rede formada por UAVs [9]

Pode-se utilizar metáforas biológicas e algoritmos bioinspirados para que um grupo de UAVs atue de forma cooperativa e com comportamento relacionado em prol de um objetivo comum. Um comportamento bio-inspirados bastante utilizado com UAVs é o comportamento de enxame [6][7][8], principalmente no tocante a busca, reconhecimento e roteamento [5].

Os enxames se comportam de forma descentralizada e com alta tolerância a falha, pois a falha de um elemento não compromete o todo. Por outro lado, temos elementos com comportamento relativamente simples que, juntos, são capazes de resolver problemas complexos [6]. Assim, temos na colaboração e coordenação de grupos de UAVs as vantagens de completar missões em um período de tempo mais curto e poder cumprir várias metas simultaneamente. Além disso, pequenas aeronaves são de difícil detecção quando comparadas a um veículo grande, como um carro, por exemplo, e, como foi colocado antes, se um UAV é danificado a missão continua sendo mantida pelos demais [9].

Segundo os artigos [7] e [10], um problema de busca com enxames de UAVs é exatamente como alocar e controlar de forma eficiente os UAVs para realizar buscas exaustivas com o menor custo energético.

No planejamento da sua distribuição em uma área, um UAV pode ser afetado por muitos fatores de risco, os quais devem ser considerados no planejamento. Dois fatores importantes nesse sentido são a habilidade de resistência da aeronave, que inclui a capacidade de manobra do UAV, velocidade de voo e velocidade de voo em altitude, e as condições ambientais da área a ser varrida pelos UAVs, como visibilidade, pressão do ar e velocidade do vento. No caso de aplicações militares, incluem-se como fatores de risco os radares, mísseis e artilharia [8].

Temos inúmeros exemplos de uso de UAVs, tanto civis quanto militares, que envolvem reconhecimento de superfícies, monitoramento de sítios arqueológicos, assistência a desastres, vigilância de fronteira, monitoramento de linhas de energia, entre outros [4].

2.2. Computação Autônoma

Com o aumento da complexidade dos sistemas computacionais, com arquiteturas orientadas a serviços e compostas por elementos de hardware e software distribuídos geograficamente, busca-se

transferir responsabilidades de gerenciamento para o próprio sistema.

Nesse cenário, novas características foram se tornando cada vez mais importantes, como dinamismo, para que as aplicações possam responder a necessidades variáveis de recursos da aplicação ao longo do tempo; complexidade, em grande parte relacionada a uma arquitetura de software relacionada a serviços; qualidade de serviço (QoS), envolvendo fatores como corretude e desempenho; emergência, que busca tratar comportamentos não antecipados, como falha de recursos e concorrência. Dessas necessidades, a IBM produziu um manifesto em 2001 onde propôs o conceito de Computação Autônoma, referente a sistemas computacionais capazes de auto-gerenciamento a partir de um conjunto de objetivos definidos [11].

Assim, temos que a computação autônoma objetiva o desenvolvimento de sistemas computacionais complexos capazes de auto-gerenciamento e adaptação a mudanças não previstas [12].

O auto-gerenciamento é o centro de um sistema autônomo e é caracterizado por quatro propriedades, chamadas self-* properties, que possibilitam esse auto-gerenciamento [13].

Essas propriedades são [14][13][11]:

- Auto-configuração (self-configuring) - o sistema deve ser capaz de realizar as auto-configurações necessárias ao seu funcionamento. Essas configurações podem ser necessárias devido a novas circunstâncias relacionadas ao seu funcionamento ou como apoio aos processos de auto-reparação, auto-otimização ou auto-proteção;
- Auto-otimização (self-optimizing) - o sistema autônomo deve buscar otimizar seu próprio funcionamento, buscando melhorar e aprimorar seu desempenho. Em síntese, o próprio sistema deve buscar seu aprimoramento;
- Auto-reparação (self-healing) - o sistema deve ser capaz de identificar falhas, assim como suas causas, e traçar soluções, que devem ser aplicadas e testadas sem necessitar de intervenção externa. Requer a identificação do problema e seu reparo imediato, sem interrupção do serviço; e
- Auto-proteção (self-protecting) - o sistema deve decidir quando e como se defender de falhas ou ataques maliciosos, assim como, antecipar possíveis falhas e se prevenir para elas antecipadamente.

Um ponto importante é que sistemas auto-gerenciáveis estão aptos a realizar ajustes apenas dentro de seu próprio escopo. Então, dado um gerenciamento autônomo de um servidor, por exemplo, apenas esse servidor pode ter seu desempenho otimizado.

Assim, temos que, sistemas de controle autônomos são projetados para funcionar bem sob incertezas significativas no sistema e ambiente por longos períodos de tempo, e eles devem ser capazes de compensar as falhas do sistema, sem intervenção externa.

Esses sistemas, muitas vezes, utilizam técnicas do campo da Inteligência Artificial (IA) para alcançar essa autonomia [4]. Segundo o artigo [5], o interesse nas pesquisas para controle e coordenação de veículos autônomos aumentou nos campos da inteligência artificial, programação dinâmica, programação linear inteira. Técnicas tradicionais de IA, como A* (A star) e suas variantes, estariam sendo aplicadas mas sem abranger cooperação

entre múltiplos veículos. Da mesma forma, algoritmos de cobertura de spanning tree trabalham bem com robô único mas não múltiplos.

O contexto dos UAVs, mais fortemente quando em ambientes externos, está inserido em um ambiente de incerteza que muda rapidamente. Nessa área, alguns estudos têm surgido sobre o controle autônomo dos UAVs, como em [4] e [15].

O artigo [15] apresenta uma técnica de avaliação que combina tecnologias de simulação para criar um método de avaliação de autonomia de um veículo aéreo não tripulado que aproveita ao máximo os ambientes virtuais visuais e simulações construtivas. Seu trabalho também apresenta nove comportamentos colaborativos de grupo que foram definidos em 2003 pela AATD (*Applied Aviation Technology Directorate* - órgão ligado às forças armadas norte-americanas responsável por sua tecnologia de aviação.). São eles, em tradução livre: Voo Cooperativo “Ver e Evitar” com Prevenção de Obstáculos e Colisão; Vingamorte / Proteção do Time; Reconhecimento / Segurança Cooperativa; Ajuste do Time a Falha de Componente; Adaptação da Rede para Assegurar Comunicações; Manter Vigilância Sobre Vários Alvos Móveis em Terreno Urbano; Estabelecer Vários Pontos Ótimos de Observação; Dupla Elevação; e Suportar Abastecimento Não-Tripulado.

Uma métrica apresentada nos artigos [15] e [4] e citada como amplamente aceita para comparar a autonomia em UAVs é a ACL (*Autonomous Control Level*), desenvolvida pelas forças aéreas norte-americanas. Ela relaciona dez níveis de autonomia em UAV, indo do Remotamente Guiado, como menor grau de autonomia, até o nível mais complexo que é o dos Enxames Plenamente Autônomos.

Neste trabalho, a relevância da computação autônoma está intimamente ligada à auto-otimização do comportamento em enxame dos UAVs, assim como a auto-configuração para busca no caso de falha de um ou mais UAVs.

2.3. Computação Bioinspirada e Comportamento de Enxame

Encontra-se na natureza uma série de exemplos de funcionamento de sistemas, compostos por elementos menores que cooperam em prol do todo, e de como solucionar problemas, dos mais simples aos mais complexos.

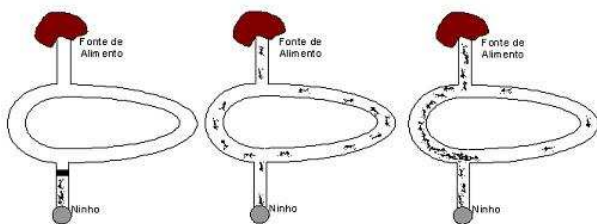


Figura 3 Ilustração sobre experimento de coleta de alimento entre formigas e escolha do menor caminho
 [Fonte: <http://www.lvcon.computacaonatural.com.br/tema?tema=4>].

No caso clássico das colônias de formigas [16], elas exploram inicialmente as fontes de comida de melhor qualidade ou que estão mais próximas da colônia para, só depois, buscar as fontes mais distantes e de menor qualidade. Na Figura 3, pode-se verificar um experimento onde um grupo de formigas está inicialmente preso e após a saída existe uma bifurcação, com um caminho longo e um

caminho curto, ambos levando à mesma fonte de comida. Quando as formigas são liberadas, algumas seguem pelo caminho mais curto e outras pelo caminho mais longo. Com o tempo, quanto mais formigas seguirem pelo caminho mais curto, maior a liberação de feromônios nesse caminho e maior a quantidade de formigas que tenderá a segui-lo. Esse comportamento pode ser utilizado para estimar a menor rota a ser seguida entre dois pontos.

O trabalho [2] nos fornece um resumo que relaciona os princípios biológicos com as suas aplicações na área de rede de computadores. A Tabela 1 oferece uma visão geral dessas aplicações.

Tabela 1 Categorização de fenômenos biológicos e algoritmos de rede que imitam esses conceitos [2]

Princípio biológico	Campo de aplicação em redes
Inteligência de enxame e insetos sociais	Busca distribuída e otimização; roteamento em redes de computadores, especialmente MANETs ¹ , WSNs e redes overlay; alocação de tarefas e recursos.
Sincronização de vagalumes	Sincronização de clock totalmente distribuída e robusta.
Sistema ativador-inibidor	Auto-organização de sistemas autônomos; coordenação distribuída; adaptação contínua de parâmetros de sistemas em ambientes altamente dinâmicos.
Sistema imunológico artificial	Segurança de rede; detecção de anomalias e mau comportamento.
Disseminação epidêmica	Distribuição de conteúdo em redes de computadores (por exemplo, nas DTNs); redes overlay; análise de propagação de worms e vírus na internet.
Redes de sinalização celular	Controle e coordenação em sistemas massivamente distribuídos; programação de redes centralizadas de sensores e atuadores.

Algoritmos bioinspirados podem ser usados tanto para problemas de otimização, quanto exploração e mapeamento, ou reconhecimento de padrões [2]. Uma das abordagens de algoritmos bioinspirados aplicados a redes de computadores, também visto na Tabela 1, é a inteligência de enxame. Ela é baseada na observação do comportamento coletivo de sistemas descentralizados e auto-organizados, como colônias de formigas ou enxames de abelhas [2][17]. As principais características, que são ao menos parcialmente compartilhadas pelos membros dessa classe de algoritmos, são o uso de uma metáfora natural, inerente paralelismo, natureza estocástica, adaptabilidade, e utilização de realimentação positiva [18].

Os algoritmos que utilizam inteligência de enxame basicamente imitam métodos existentes na natureza para direcionar a busca de uma solução ótima e usam um conjunto de soluções a cada iteração, ao invés de apenas uma [19].

Segundo o artigo [20], a inteligência de enxame tem cinco princípios básicos. O primeiro é o princípio da proximidade, onde a população deve ser capaz de executar cálculos de tempo e espaço simples. O segundo é o princípio da qualidade, onde a população

¹ A MANET (*mobile ad hoc network*) é uma rede móvel que se estabelece conforme a necessidade.

deve ser capaz de responder satisfatoriamente a elementos do ambiente. O terceiro é o princípio da resposta diversificada, onde a população não deve comprometer suas atividades em canais excessivamente estreitos, sendo capaz de manter sua diversidade. O quarto é o princípio da estabilidade, onde a população não deve mudar seu comportamento toda vez que o ambiente mudar. O quinto é o princípio da adaptabilidade, pelo qual a população deve ser capaz de mudar seu comportamento quando o custo computacional valer a pena.

Se um problema de otimização possuir apenas uma solução ótima, as populações geradas pelos algoritmos baseados em inteligência de enxame tenderão a convergir para a solução ótima. Caso tenham múltiplas soluções ótimas, esse tipo de algoritmo pode ser usado para capturá-las em sua população final [19].

Algumas soluções podem trabalhar em conjunto com Algoritmos Genéticos (no inglês, Genetic Algorithm - GA) com heurísticas que dependem do domínio para melhorar os resultados. Os Algoritmos Genéticos são baseados em seleção natural e recombinação genética e funcionam escolhendo soluções da população corrente e aplicando operações genéticas, como mutações e cruzamentos, para criar uma nova população [19].

Uma característica que pode se mostrar difícil ao trabalhar com inteligência de enxame é que, pelo seu comportamento probabilístico, uma solução executada uma segunda vez não terá exatamente o mesmo comportamento que a primeira execução, não sendo possível, portanto, reproduzir igualmente cada simulação. No entanto, segundo o artigo [21], as soluções mais usadas para ambientes em tempo real são os métodos probabilísticos, como Ant Colony e Particle Swarm Optimization. Outros algoritmos com inteligência de enxame vistos na literatura são o Bee Colony (BC) e o Wolf Search Algorithm (WSA).

3. TRABALHOS RELACIONADOS

Alguns dos trabalhos identificados durante o levantamento bibliográfico apresentaram maior relação com a presente pesquisa. Foram selecionados principalmente trabalhos que estudassem posicionamento de UAVs e comportamento bioinspirado.

O trabalho [22] apresenta uma abordagem para detecção cooperativa de incêndios por uma frota de UAVs heterogêneos. Os objetivos são determinar a posição dos alarmes de possíveis incêndios e reduzir o número de falsos alarmes por meio da cooperação entre os UAVs. Ele aborda algoritmos de detecção de incêndio utilizando visão, controle de alarme e fusão de dados. Foram realizados experimentos com pequenos fogos controlados. O objetivo de usar múltiplos UAVs foi de melhorar a precisão das detecções e diminuir o número de falsos alarmes. No experimento cujos resultados são apresentados, foram utilizados dois UAVs, um portando câmera de vídeo visual e o outro uma câmera infravermelha. O posicionamento dos UAVs é feito a partir de uma distribuição de probabilidade, o que não se adequa ao desejado no presente trabalho.

O artigo [6] propõe um novo comportamento de busca onde se divide a área de busca, e quando combinado com soluções de enxame anteriores, constitui um problema de otimização de como melhor atribuir soluções de enxame para uma topologia complexa. Ele aborda dois algoritmos de busca cooperativa descentralizada que utilizam inteligência de enxame, o Parallel Path Search e o SWEEP Protocol e propõe que a área de busca seja dividida em sub-regiões de forma a usar esses dois algoritmos em pedaços

separados do mapa original de busca de acordo com o melhor em cada divisão do mapa. Devido às limitações desses dois algoritmos, com relação a terreno e custo computacional, é proposta a ideia de adicionar UAVs sentinela, o que gera um novo problema de otimização visto que se deve buscar a melhor forma de particionar o mapa considerando os UAVs disponíveis, tempo de busca, e custos de comunicação, coordenação e processamento. Essa busca melhorada baseada em sentinela ainda estava em desenvolvimento quando da publicação do artigo.

O artigo [1] aborda dois pontos bio-inspirados com Micro Air Vehicles (MAVs), o vôo de insetos e a detecção de borda e reconhecimento de forma usado por abelhas e outros insetos. Esse trabalho apresenta um repertório de módulos comportamentais que podem ser controlados por meio de máquinas de estados finitos (do inglês, Finite State Machines - FSM); e detectores de movimentos básicos (do inglês, Elementary Movement Detectors - EMD) combinados com um algoritmo de detecção de borda centralizada ou espaçada para produzir melhor informação na detecção de objetos. O artigo utiliza máquinas de estados finitos para projetar o comportamento dos MAVs em ambiente indoor, onde os status descrevem o tipo de comportamento (e.g. Left Turn) e as transições descrevem as condições necessárias para mover de um estado a outro (e.g. Turn Complete). Outro ponto do artigo é combinar detecção de borda com informação de movimento captada por EMDs, de forma que as informações de EMD possam ser usadas como filtro para remover o ruído na identificação das bordas. O foco do artigo é o voo autônomo de MAVs em ambientes indoor buscando autonomia no trajeto e desvio de obstáculos.

O trabalho [23] apresenta uma proposta de determinação de rota para UAV utilizando o Ant Colony Optimization (ACO). A ideia principal é dividir a área em grids e, tendo um ponto inicial e um ponto final definidos, utilizar o ACO para determinar a menor rota. Este trabalho não se aplica à presente pesquisa por tratar com destino predeterminado.

O artigo [24] apresenta o resultado de simulações com dois conjuntos de algoritmos buscando identificar o melhor caminho para um UAV com câmera de vídeo onboard para percorrer de forma a cobrir um maior espaço possível. Cada conjunto teve três algoritmos implementados, baseados nas ideias de *Local Hill Climbing* (LHC), Convolução e Algoritmos Evolucionários, sendo um grupo trabalhando sem área limite definida e outro com área limite definida. A ideia era determinar o algoritmo que gerava o caminho mais eficiente em termo de cobertura e com menor tempo de execução. Seus testes apontaram os algoritmos LHC com Convolução como os mais eficientes na maioria dos testes propostos. Esse trabalho apresentou boas propostas para geração de caminho para o UAV mas não trata do comportamento cooperativo de vários UAVs, como é o desejado no presente trabalho.

A pesquisa apresentada no artigo [7] apresenta uma proposta de determinação de rota para grupos de UAV utilizando o Ant Colony System (ACS). Sua particularidade consiste na adoção de pontos de passagem que são utilizados pelos UAVs para planejar seus caminhos a partir deles (técnica usada principalmente por UGVs). A seleção dos pontos de passagem é a parte que utiliza o ACS, assim cada UAV se desloca até o ponto de passagem mais próximo a ele, sem haver repetição na escolha, seguindo um sentido predeterminado, até cada um estar em um ponto final. Com isso, os UAVs seguem uma rota tão em linha reta quanto possível. Esse trabalho se mostrou bastante relevante, tangenciando o objetivo da presente pesquisa. No entanto, mostra-se voltado para detecção de

alvos fixos, buscando identificar e destruir sistemas de defesa. O trajeto mostrado como exemplo no artigo não se mostraria eficiente no caso de alvos menores ou em movimento.

O artigo [39] apresenta uma proposta para determinar rotas de reconhecimento para UAVs, considerando existência dos alvos e pontos de ameaças aos UAVs, e utilizando o Particle Swarm Optimization (PSO) como base para seu algoritmo.

O trabalho [3] propõe um algoritmo baseado em Algoritmos Evolucionários para encontrar um caminho ótimo para um ou um grupo de UAVs, em cenários realistas e de risco. Ele considera constraints como os limites do mapa a ser percorrido e o autonomia dos UAVs com relação a combustível. O melhor caminho proposto busca o menor percurso, a menor probabilidade de ser abatido no percurso, menor altitude de voo para diminuir o consumo de combustível e a menor probabilidade de detecção por radar. Este trabalho não se aplica à presente pesquisa pois trata com pontos de início e de fim predeterminados.

A presente pesquisa utiliza alvos tanto fixos quanto em movimento e pretende testar a eficiência e adequação de um dado algoritmo bioinspirado.

4. PROPOSTA

Na seleção dos algoritmos, foram considerados alguns pontos que poderiam confirmar sua adequação ou que poderiam indicar que determinado algoritmo não era adequado. No caso do PSO, ele leva em consideração o melhor desempenho que cada um dos indivíduos teve até o momento. Em problemas como localização de mínimo de uma função ou caixeiro viajante, ele seria mais adequado do que no presente problema de busca, onde não se conhece a localização dos alvos. Não sabendo essa localização não é possível identificar uma melhor posição ou um ótimo atual. A primeira abordagem heurística para solução do problema foi o uso do *Particle Swarm Optimization*. O algoritmo foi alterado para se comportar como um bando em busca de um dado alvo. Os algoritmos com PSO mais comumente encontrados são referentes ao problema do caixeiro viajante e ponto máximo ou mínimo.

No ACO, o caminho de cada formiga fica registrado por uma trilha de feromônios e à medida que mais formigas acham um menor caminho, ele tende a ser reforçado.

No BC, abelhas batedoras exploram uma região e quando acham o alimento retornam à colmeia com o pólen e informam a direção do alimento, sua distância e qualidade.

No WSA, os lobos caçam em matilhas, permanecendo silenciosos e furtivos quando nessa caça. Cada lobo caça de forma independente, lembrando seu percurso, e apenas se une a outro da matilha se esse se encontra em melhor posição quanto à presa. Dessa forma, a comunicação a longo alcance é praticamente eliminada entre eles.

Verificando essas características, o WSA se mostrou mais adequado ao problema proposto, pois matilhas de lobos formam grupos com comportamento mais próximo ao desejado pelo grupo de UAVs - grupos pequenos e autônomos - do que o encontrado em formigas e abelhas. Por outro lado, o PSO foi descartado pela característica de ótimo atual que não se encaixa com esse tipo de busca.

Foi utilizado o Matlab para verificação e adaptação do algoritmo PSO² e simulação do comportamento dos UAVs na busca dos

alvos. Foi escolhido o PSO para avaliação inicial por se tratar de um algoritmo já estabelecido e conhecido na literatura. Os demais algoritmos foram comparados segundo suas características, as quais foram colhidas através da pesquisa bibliográfica.

Foi utilizado uma grid n x m, com número de UAVs configurável, tendo o valor 5 (cinco) como padrão no caso do parâmetro não ser passado. Nesse cenário inicial foi usado apenas um alvo fixo. Tanto o alvo quanto os UAVs foram dispostos aleatoriamente nessa grid. À medida que as iterações iam acontecendo os UAVs se movimentavam como um bando seguindo a mesma orientação na movimentação. Quando um dos UAVs localizava o alvo, os demais seguem ao seu encontro, confirmando a localização.

Na sequência de figuras a seguir (Figura 4 a Figura 8) tem-se algumas telas salvas no Matlab que mostram quadros intermediários da movimentação dos UAVs. Nelas, os UAVs são marcados com 'o' e o alvo é marcado com '*'.

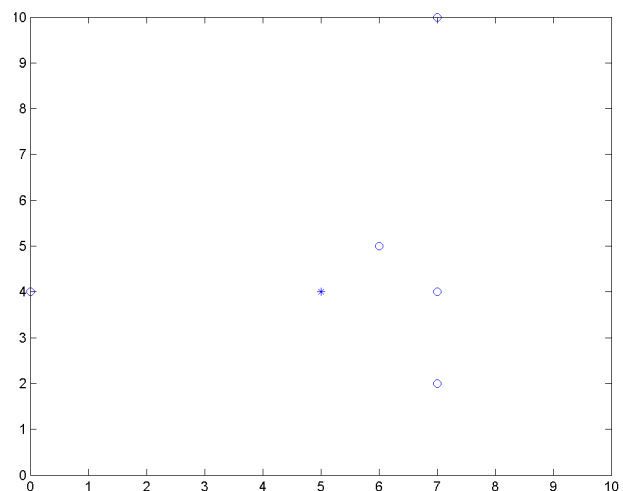


Figura 4 Tela A mostrando alvo (asterisco azul) e movimentação dos UAVs (círculos azuis)

Na Figura 4, vemos a disposição inicial, gerada aleatoriamente, dos UAVs e do alvo. O objetivo é que os UAVs o localizem dentro do tempo desejado que, nesse caso, é dado pelo número estipulado de iterações.

Até que o alvo seja localizado, os UAVs seguem uma trajetória comum, varrendo o terreno em busca do alvo. As Figuras 5 a 8 mostram algumas posições intermediárias capturadas da movimentação dos UAVs em direção ao alvo, bem como uma captura do posição final. Na simulação, quando o primeiro UAV localiza o alvo, os demais seguem para sua posição para confirmar a localização.

À medida que as iterações são realizadas, os UAVs movem-se seguindo o mesmo padrão de deslocamento, convergindo para o mesmo local caso o alvo seja localizado.

² Códigos disponíveis em <https://github.com/patriciapaula/st-swarm-uav>

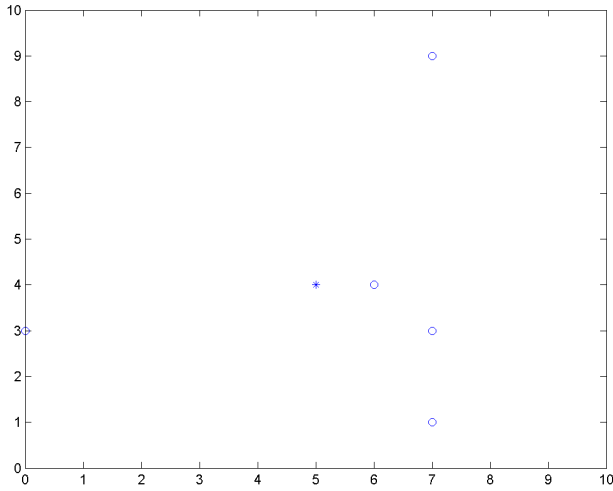


Figura 5 Tela B mostrando alvo e movimentação dos UAVs

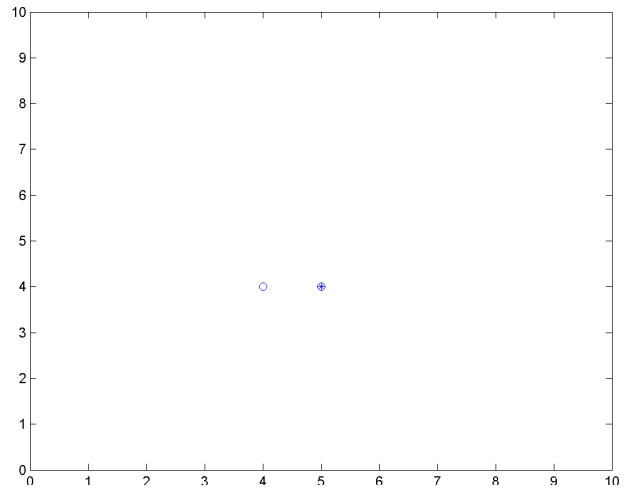


Figura 8 Tela E mostrando alvo e movimentação dos UAVs

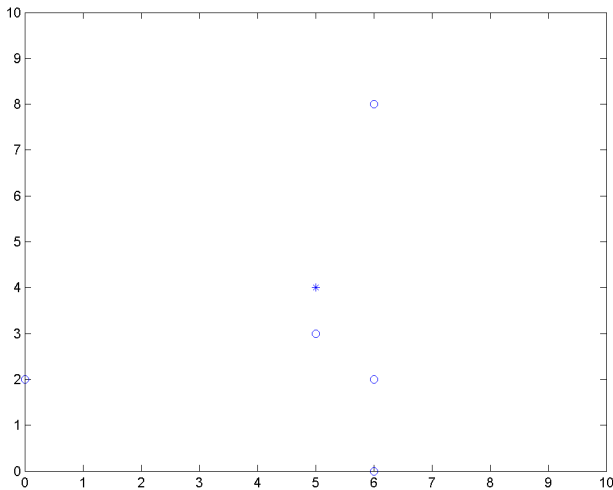


Figura 6 Tela C mostrando alvo e movimentação dos UAVs

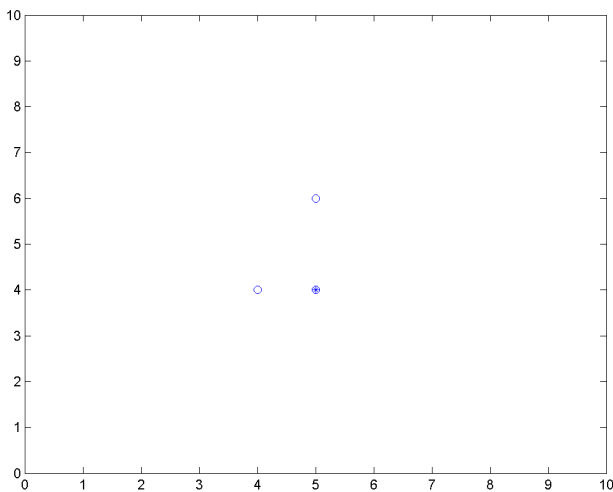


Figura 7 Tela D mostrando alvo e movimentação dos UAVs

Os primeiros testes realizados atingiram o objetivo em aproximadamente 65% das execuções. Ainda estão sendo verificadas outras configurações do PSO, visando atingir uma melhora nos resultados. No momento, também estão sendo feitos testes com outros algoritmos cujas características convergem para a solução deste problema.

Tabela 2 Resumo sobre algoritmos bioinspirados

Algoritmo	Descrição
Ant System	Baseado no forrageamento das formigas. Uma formiga segue um caminho aleatório para buscar alimento e quando volta ao ninho deposita no caminho algum feromônio, em quantidade variável, que serve como uma trilha para buscas futuras [2][18]. Quando uma formiga encontra uma trilha de feromônio já existente, ela pode detectar e decidir seguir por esse caminho com uma maior probabilidade, reforçando, assim, a trilha com o seu próprio feromônio. Dessa forma, quanto mais formigas seguirem por um caminho, mais atraente às demais ele se torna [18].
Bee Colony (BC)	Baseado no forrageamento das colônias de abelhas. As abelhas batedoras saem e exploram a região em busca de uma fonte de alimento. Quando acham, elas retornam à colmeia, depositam o néctar e pólen e seguem para transmitir e compartilhar a descoberta com os demais membros da colmeia, informando a direção do alimento, sua distância da colmeia e sua qualidade. Novas abelhas são recrutadas de forma proporcional à quantidade de alimento informado pelas abelhas batedoras [16][19].
Particle Swarm Optimization (PSO)	Baseado na vida artificial de grupos como bandos de aves e cardumes de peixes. Tem-se um conjunto de indivíduos, chamados partículas e se leva em consideração o melhor desempenho que cada um desses indivíduos teve até o momento, assim como

	o melhor desempenho dos indivíduos vizinhos para decidir os próximos passos [16]. O compartilhamento de informações entre os elementos é vital nesse algoritmo.
Wolf Search Algorithm (WSA)	Baseado na busca dos lobos por alimentos e sua sobrevivência em evitar ameaças. Os lobos caçam juntos em matilhas, formando um único núcleo familiar, mas permanecem em silêncio e furtivos quando na caça. Cada um caça de forma independente, lembrando seu percurso, e apenas se une a outro da matilha se esse se encontra em melhor posição quanto à presa. Dessa forma, a comunicação a longo alcance é praticamente eliminada. Nesse processo de caça, eles evitam potenciais ameaças, como humanos ou outros predadores animais

5. CONCLUSÃO

Dentro do objetivo proposto de permitir que um grupo de aeronaves não tripuladas possam se juntar e ajudar na busca e resgate de pessoas, o presente trabalho aponta para o uso de propostas bioinspiradas de otimização das buscas. No entanto, destaca-se como principal contribuição a análise de resultados da adaptação do PSO para esse problema. Como segunda contribuição, a seleção via pesquisa bibliográfica, considerando as particularidades dos algoritmos mais conhecidos na literatura, de um algoritmo mais viável para o cenário proposto que serviria de base para outros problemas similares. No caso, este algoritmo seria o *Wolf Search Algorithm*.

Como trabalho futuro, pretende-se continuar a alteração do PSO, testando variações em sua configuração, visando um melhor resultado final. Bem como adaptar o WSA para a proposta de busca de alvos fixos ou em movimento, com posição desconhecida. Assim como sua comparação com os resultados já obtidos até o momento. Todos os dados desta pesquisa estão sendo disponibilizados no github para que demais pesquisadores possam efetuar suas simulações e verificar os dados e informações obtidas.

6. AGRADECIMENTOS

Agradeço à entidade a qual faço parte por ter gentilmente apoiado e colaborado para a realização dessa pesquisa.

7. REFERÊNCIAS

- [1] Gerke, P. et al., 2011. BioMAV: bio-inspired intelligence for autonomous flight. *International Micro Air Vehicle Conference and Flight Competition*.
- [2] Dressler, F. & Akan, O., 2009. A survey on bio-inspired networking. *Computer Networks*, 54.
- [3] Cruz, J. et al., 2008. Evolutionary path planner for UAVs in realistic environments. *10th annual conference on Genetic and evolutionary computation*, pp. 1477-1484.
- [4] Chen, H., Wang, X. & Li, Y., 2009. A Survey of Autonomous Control for UAV. *International Conference on Artificial Intelligence and Computational Intelligence*.
- [5] Nigam, N., Bieniawski, S., Kroo, I. & Vian, J., 2012. Control of Multiple UAVs for Persistent Surveillance: Algorithm and Flight Test Results. *IEEE Transactions On Control Systems Technology*, Volume 20, pp. 1236-1251.
- [6] McCune, R. & Madey, G., 2013. Agent-Based Simulation of Cooperative Hunting with UAVs. *Symposium on Agent Directed Simulation (ADS '13/SpringSim 2013)*.
- [7] Wei, L. & Wei, Z., 2009. Path Planning of UAVs Swarm Using Ant Colony System. *International Conference on Natural Computation*.
- [8] Li, S., Sun, X. & Xu, Y., 2006. Particle Swarm Optimization for Route Planning of Unmanned Aerial Vehicles. *IEEE International Conference on Information Acquisition*.
- [9] Sathyaraj, B., Jain, L., Finn, A. & Drake, S., 2008. Multiple UAVs path planning algorithms: a comparative study. *Fuzzy Optimization and Decision Making*, Volume 7, pp. 257 - 267.
- [10] Basilico, N. & Carpin, S., 2012. Online patrolling using hierarchical spatial representations. *IEEE International Conference on Robotics and Automation*.
- [11] Correa, S. & Cerqueira, R., 2009. Computação autônoma: Conceitos, Infra-estruturas e Soluções em Sistemas Distribuídos. *27o SBRC - Livro Texto dos Minicursos*, pp. 151-198.
- [12] Ribeiro, L. & Castro, M., 2010. BiO4SeL: A Bio-Inspired Routing Algorithm for Sensor Network Lifetime Optimization. *17th International Conference on Telecommunications*.
- [13] Ribeiro, L., 2009. BiO4SeL: Uma Abordagem Baseada em Colônia de Formigas para a Otimização do Tempo de Vida de Redes de Sensores Sem Fio.
- [14] IBM, 2005. An architectural blueprint for autonomic computing - 3a Ed.
- [15] Sholes, E., 2007. Evolution of a UAV Autonomy Classification Taxonomy. *Aerospace Conference*.
- [16] Castro, L., 2010. Computação Natural - Uma Jornada Ilustrada. São Paulo: Livraria da Física.
- [17] Bitam, S., Batouche, M. & Talbi, E., 2010. A survey on bee colony algorithms. *IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW)*, pp. 1-8.
- [18] Dorigo, M., Maniezzo, V. & Colomi, A., 1996. Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics-Part B Cybernetics*, pp. 29-41.
- [19] Pham, D. et al., 2006. The Bees Algorithm – A Novel Tool for Complex Optimisation. *2nd International Virtual Conference on Intelligent Production Machines and Systems*, pp. 454-459.
- [20] Kennedy, J. & Eberhart, R., 1995. Particle Swarm Optimization. *IEEE International Conference on Neural Networks*, pp. 1942 - 1948.
- [21] Souissi, O. et al., 2013. Path Planning: A 2013 Survey. *5th IESM Conference*, Outubro.
- [22] Merino, L., Caballero, F., Dios, J. & Ollero, A., 2005. Cooperative Fire Detection Using Unmanned Aerial Vehicles. *Int. Conf. Robotics and Automation (ICRA 05)*.

- [23] Zhang, C., Zhen, Z., Wang, D. & Li, M., 2010. UAV path planning method based on ant colony optimization. *Chinese Control and Decision Conference*, Maio, pp. 3790-3792.
- [24] Lin, L. & Goodrich, M. A., 2009. UAV Intelligent Path Planning for Wilderness Search and Rescue. *International Conference on Intelligent Robots and Systems*, pp. 709-714.
- [25] Bao, Y., Fu, X. & Gao, X., 2010. Path Planning for Reconnaissance UAV Based on Particle Swarm Optimization. *International Conference on Computational Intelligence and Natural Computing (CINe)*, pp. 28-32.